

REMARKS

Herein, the "Action" or "Office Action" refers to the Office Action dated 11/30/2005.

Applicant respectfully requests reconsideration and allowance of all of the claims of the application. Claims 1-42 and 45-46 are presently pending. Claims amended herein are 10 and 29. Claims withdrawn or cancelled herein are 43-44. New claims added herein are none.

Election/Restriction

In the Action, the Office states:

Election/Restrictions

3. Restriction to one of the following inventions is required under 35 U.S.C. 121:

I. Claims 1-42, 45-46, drawn to a kernel emulator comprising translating instructions, addresses, and arguments from non-native to native code, classified in class 703, subclass 23.

II. Claims 43-44, drawn to a kernel emulator comprising a target platform determiner, an instruction type detector, a translator selector, classified in class 703, subclass 23.

4. Inventions I and II are related as subcombinations disclosed as usable together in a single combination. The subcombinations are distinct from each other if they are shown to be separately usable. In the instant case, invention II has separate utility such as determining the target platform, determining the type of non-native instructions, and selecting a translator capable of translating the instructions. See MPEP § 806.05(d).

5. Because these inventions are distinct for the reasons given above and have acquired a separate status in the art because of their recognized divergent subject matter, restriction for examination purposes as indicated is proper.

6. During a telephone conversation with Mr. Kasey Christie on 11/11/04 a provisional election was made without traverse to prosecute the invention of Group I, Claims 1-42, 45-46. Affirmation of this election must be made by applicant in replying to this Office action. Claims 43-44 withdrawn from further consideration by the examiner, 37 CFR 1.142(b), as being drawn to a non-elected invention.

Applicant confirms the election of claims 1-42 and 45-46. Accordingly, Applicant withdraws claims 43 and 44 from consideration here.

Informalities

Specification

The Office indicates that lines 16 of page 23 should read "within" instead of "with in." Applicant amends herein to correct accordingly.

Claim Objections

The Office asks for small informalities to claims 10 and 42 to be corrected. Applicant amends herein to correct accordingly.

Claim Interpretation

The Office indicates:

Claim Interpretation

12. Claims 9 and 29, recite a "shared-memory manager configured to synchronize a native shared data structure with a non-native shared data structure". It was unclear as to what "synchronize" meant. The specification states, "The address conversion (between native and non-native addressing formats) is done during synchronization" (page 23, lines 13-15). From this statement in the specification, it was determined that synchronization is the address conversion of a non-native address to a native address.

Applicant submits that "synchronize" as used in these claims (claims 9 and 29) does indeed include address conversion. However, synchronization is not limited to just address conversion. It may include other data format conversions.

The two data structures are "synchronized" when the data in one of the data structures is accessible in the other data structure. This includes, for example, address conversion, instruction conversion, argument-format conversion, word-size conversion, and other conversions described in the Application and other similar conversions consistent with a native to/from non-native conversion.

Substantive Claim Rejections

Claim Rejections under §§ 102 and 103

The Office rejects all of the pending claims (1-42 and 45-46) under §102 and/or §103. For the reasons set forth below, the Office has not shown that one or more of the cited references anticipate the rejected claims. For the reasons set forth below, the Office has not made a *prima facie* case showing that the rejected claims are obvious (under §103). Accordingly, Applicant respectfully requests that the rejections be withdrawn and the case be passed along to issuance.

The Office's rejections are based upon the following references:

- **Scalzi:** *Scalzi et al.*, US Patent No. 5,560,013 (issued 9/24/1996);
- **Franz:** Michael Franz, "Emulating an Operating System on Top of Another" Software – Practice and Experience. Vol. 23, No. 6, June 1993, pp. 677-692);
- **Duvall:** *Duvall et al.*, US Patent No. 4,742,447 (issued 5/3/1988);
- **McCoy:** *McCoy et al.*, US Patent No. 5,036,484 (issued 7/30/1991).

Overview of the Application

The Application describes a technology facilitating the operation of non-native program modules within a native computing platform. More particularly, it describes a technology facilitating the interoperability of native and non-native program modules within a native computing platform.

Specifically, this technology involves an emulation of the kernel of the non-native operating system. Instead of interacting with the native kernel of the native computing platform, the non-native program modules interact with a non-native

421 West Riverside, Suite 500
 Spokane, WA 99201
 P: 509.324.9256
 F: 509.323.8979
 www.lee&hayes.com
lee & hayes

1 kernel emulator. This emulator handles the necessary conversions and translations.
2 With this non-native kernel emulation, native and non-native program modules are
3 interoperable. Except for the kernel emulator, none of the program module (native
4 or non-native) and none of the other portions of the native computing platform are
5 aware of the emulation. The computing environment and other program modules
6 appear to be non-native to the non-native program modules. Likewise, the non-
7 native program modules appear to be native to the computing environment and the
8 native program modules.

9 10 Cited References

11 The Office cites **Scalzi** as its reference for is anticipation-based rejections
12 and the primary references in many of its obviousness-based rejections. The
13 Office cites **Duvall** as its primary reference for some of its obviousness rejections
14 and a secondary reference in other obviousness-based rejections. In addition, the
15 Office cites **Franz** and **McCoy** as secondary references in some of its
16 obviousness-based rejections.

17 18 Scalzi

19 Scalzi describes a method of utilizing large virtual addressing in a target
20 computer to implement an instruction set translator (IST) for dynamically
21 translating the machine language instructions of an alien source computer into a
22 set of functionally equivalent target computer machine language instructions,
23 providing in the target machine, an execution environment for source machine
24 operating systems, application subsystems, and applications.

1 The target system provides a unique pointer table in target virtual address
2 space that connects each source program instruction in the multiple source virtual
3 address spaces to a target instruction translation which emulates the function of
4 that source instruction in the target system. The target system stores the translated
5 executable source programs by actually storing only one copy of any source
6 program, regardless of the number of source address spaces in which the source
7 program exists.

8 The target system manages dynamic changes in the source machine storage,
9 accommodating the nature of a preemptive, multitasking source operating system.
10 The target system preserves the security and data integrity for the source programs
11 on a par with their security and data integrity obtainable when executing in source
12 processors (i.e. having the source architecture as their native architecture). The
13 target computer execution maintains source-architected logical separations
14 between programs and data executing in different source address spaces--without a
15 need for the target system to be aware of the source virtual address spaces.

16
17 Duvall

18 Duvall describes a method for accessing information in a page segmented
19 virtual memory data processing system in which virtual machines running UNIX
20 type operating systems are concurrently established, and in which a memory
21 manager controls the transfer of information between primary and secondary
22 storage devices in response to the occurrence of page faults. The method
23 establishes a plurality of data structures in a dynamic manner in response to a
24 Supervisor call to "map" a file.

1 The mapping process assigns a new segment of virtual memory to the
2 mapped file and correlates, in one data structure, the virtual address of each page
3 of data in the new segment to a disk file address where that page is actually stored.
4 A UNIX system call by an application program for a specific virtual page is
5 handled by the page fault hanger, and not the UNIX kernel, since the application
6 can supply the real address of the page on the disk file from the data structure that
7 was created by the mapped page range Supervisor call. Simple load and store type
8 of instructions are employed for the data transfer, which avoids much of the
9 overhead that normally accompanies conventional UNIX read and write system
10 calls to the storage subsystem.

11
12
13 Franz

14 As its name implies, **Franz** discusses the emulation of one operating
15 system on top of another operating system. **Franz** describes the design of an
16 operating-system emulator. This software interface provides the services of one
17 operating system (e.g., Oberon) on a machine running a different operating system
18 (e.g., Macintosh), by mapping the functions of the first onto equivalent calls to the
19 second.

20
21 McCoy

22 **McCoy** describes a system for emulating the operation of a terminal
23 connected to a host computing system while retaining the ability to utilize personal
24 computer application programs resident in the personal computer by utilizing a
25

1 personal computer/host terminal emulation program which conducts an analysis of
2 host data and keystrokes to identify personal computer commands and calls the
3 appropriate resident application program in response to such commands.
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324.9256
F: 509.323.8979
www.leeandhayes.com
lee & hayes

Serial No.: 09/847,535
Atty Docket No.: MS1-665us
RESPONSE TO OFFICE ACTION DATED 11/30/2004

20

0403051506 Ct:MS1-01665us MS1-665us n.01.doc
atty: Kasey C. Christia

Anticipation Rejections

Anticipation Rejections Based upon Scalzi

The Office rejects claims 1, 3-6, 9-13, 15-17, 19-28, 34, 37-40, 42, and 45-46 under USC § 102(b) as being anticipated by Scalzi. Applicant respectfully traverses the rejections of these claims. Based on the reasons given below, Applicant asks the Office to withdraw its rejections of these claims.

Claims 1, 3, 4, 13, 15, 16, 34, 40, and 45

In the Action, the Office provides the following reasoning for rejecting these claims:

14. ~~Claims 1,3-6,9-13,15-17,19-28, 34,37-40,42,45-46~~ are rejected under 35 U.S.C. 102(b) as being anticipated by Scalzi et al (U.S. Patent Number 5,560,013), herein referred to as Scalzi.

15. As to ~~Claims 1,3,4,13,15,16,34, 40 and 45~~, Scalzi teaches: a kernel emulator for non-native program modules, the emulator comprising: an interceptor configured to intercept kernel calls from non-native program modules (Figure 1, element 102 and description); a call-converter configured to convert non-native kernel calls intercepted by the interceptor into native kernel calls (Figure 1, element 103 and description); wherein the call-converter comprises: an instruction-translator configured to translate non-native CPU instructions into native CPU instructions (Figure 1, element 103 and description); an address-translator configured to translate addresses from non-native length into native length (Figure 3 and description, column 21, lines 42-48).

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324.9256
F: 509.323.8979
www.lee&hayes.com

lee & hayes
ATTORNEYS AT LAW

1 All of these rejected claims recite kernel emulation¹ and operating on or in
2 response to kernel calls. Scalzi never discloses a call to a kernel of an operating
3 system. Furthermore, Scalzi never discloses emulating a kernel of an operating
4 system. Further still, Scalzi never even mentions a kernel of an operating system.

5 Moreover, many of these claims² recite an interception of a kernel call.
6 Since Scalzi never mentions kernel calls, it is logical that is never discloses
7 intercepting such calls.

8 While Scalzi does appear to disclose instructions conversion, it fails to
9 disclose kernel emulation, kernel calls, and interception of such kernel calls. As
10 shown above, Scalzi does not disclose all of the claimed elements and features of
11 these claims. Accordingly, Applicant asks the Office to withdraw its rejection of
12 these claims.

13 14 Claims 2-13

15 These claims ultimately depend upon independent claim 1. As discussed
16 above, claim 1 is allowable.

17 In addition to its own merits, each of these dependent claims is allowable
18 for the same reasons that its base claim is allowable. Applicant submits that the
19 Office withdraw the rejection of each of these dependent claims because its base
20 claim is allowable.

21
22
23
24 ¹ E.g., "kernel emulator" in claims 1, 40 and 45, "emulating a kernel" in claim 13, and "emulating
a non-native kernel" in claim 34

25 ² E.g., independent claims 1, 13, and 45

Claims 14-28

These claims ultimately depend upon independent claim 13. As discussed above, claim 13 is allowable.

In addition to its own merits, each of these dependent claims is allowable for the same reasons that its base claim is allowable. Applicant submits that the Office withdraw the rejection of each of these dependent claims because its base claim is allowable.

Claims 35-39

These claims ultimately depend upon independent claim 34. As discussed above, claim 34 is allowable.

In addition to its own merits, each of these dependent claims is allowable for the same reasons that its base claim is allowable. Applicant submits that the Office withdraw the rejection of each of these dependent claims because its base claim is allowable.

Claims 41 and 42

These claims ultimately depend upon independent claim 40. As discussed above, claim 40 is allowable.

In addition to its own merits, each of these dependent claims is allowable for the same reasons that its base claim is allowable. Applicant submits that the Office withdraw the rejection of each of these dependent claims because its base claim is allowable.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324.9256
F: 509.323.8979
www.leeandhayes.com

lee & hayes

Claim 46

This claim ultimately depends upon independent claim 45. As discussed above, claim 45 is allowable.

In addition to its own merits, this dependent claim is allowable for the same reasons that its base claim is allowable. Applicant submits that the Office withdraw the rejection of this dependent claim because its base claim is allowable.

Obviousness Rejections**Lack of *Prima Facie* Case of Obviousness (MPEP § 2142)**

Applicant disagrees with the Office's obviousness rejections. Arguments presented herein point to various aspects of the record to demonstrate that all of the criteria set forth for making a *prima facie* case have not been met.

Based upon Scalzi and Franz

The Office rejects claims 2 and 14 under USC § 103(a) as being unpatentable over Scalzi in view of Franz. Applicant respectfully traverses the rejection of this claim. Applicant asks the Office to withdraw its rejection of this claim.

Claims 2 and 14

Claim 2 ultimately depends upon independent claim 1. As discussed above, claim 1 is allowable. Claim 14 ultimately depends upon independent claim 13. As discussed above, claim 13 is allowable.

1
2 In addition to its own merits, each of these dependent claims is allowable
3 for the same reasons that its base claim is allowable. Applicant submits that the
4 Office withdraw the rejection of each of these dependent claims because its base
5 claim is allowable.

6
7 **Based upon Scalzi and Duvall**

8 The Office rejects claims 7, 8, 18, 35, and 41 under USC § 103(a) as being
9 unpatentable over Scalzi in view of Duvall. Applicant respectfully traverses the
10 rejection of this claim. Applicant asks the Office to withdraw its rejection of this
11 claim.

12
13 **Claims 7, 8, 18, 35, and 41**

14 Claims 7 and 8 ultimately depend upon independent claim 1. As discussed
15 above, claim 1 is allowable. Claim 18 ultimately depends upon independent claim
16 13. As discussed above, claim 13 is allowable. Claim 35 ultimately depends upon
17 independent claim 34. As discussed above, claim 34 is allowable. Claim 41
18 ultimately depends upon independent claim 40. As discussed above, claim 40 is
19 allowable.

20 In addition to its own merits, each of these dependent claims is allowable
21 for the same reasons that its base claim is allowable. Applicant submits that the
22 Office withdraw the rejection of each of these dependent claims because its base
23 claim is allowable.
24
25

Serial No.: 09/847,535

Atty Docket No.: MS1-665us

RESPONSE TO OFFICE ACTION DATED 11/30/2004

25

0408051508 G:\MS1-01665us\MS1-665us m01.doc

atty: Kasey C. Christie

Based upon Duvall and McCoy

The Office rejects claims 29-33 under USC § 103(a) as being unpatentable over Duvall in view of McCoy. Applicant respectfully traverses the rejection of this claim. Applicant asks the Office to withdraw its rejection of these claims.

Claim 29

In the Action, the Office provides the following reasoning for rejecting this claim:

41. Claims 29-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Duvall and further in view of McCoy et al (U.S. Patent Number 5,036,484), herein referred to as McCoy.

42. As to Claim 29, Duvall teaches: a method comprising: if the initiating program is non-native: limiting available memory to a range that is addressable by the non-native program module (column 4, lines 43-46, column 6, lines 25-29, column 9, lines 20-25); establishing non-native a version of a shared memory data structure that may be synchronized with a native version of the same shared memory data structure (column 5, lines 45-51, column 6, lines 25-29).

43. Duvall further teaches the data in a segment of virtual memory is created as a result of an application program being run (column 5, lines 52-55). While this implies that must be some determination as to whether a program is native or non-native allowing for the segment in virtual memory to be created, Duvall does not expressly teach determining whether an initiating program module is a native or non-native.

44. McCoy teaches determining whether an initiating program module is a native or non-native (Figure 3a, element 36a, column 5, lines 40-48) in a system that emulates a host program in a PC environment and translates host data to PC format by the emulation program (column 5, lines 28-31), allowing the system to know whether to perform a function of the native system or perform a function of the non-native system which includes the translation of code (Figure 3a, element 31a and column 5, lines 40-48) since in the emulation systems of the prior art, when operating in emulation mode, the native system is incapable of performing functions other than those of the terminal which is being emulated.

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509 324-9256
F: 509 323-8979
www.lee&hayes.com

lee & hayes

Therefore, the functions of the personal computer are not available in the emulation mode (column 1, lines 32-39).

45. It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the determination of whether a program module is native or non-native as taught in Duvall with the method of determining whether a program module is native or non-native as taught by McCoy since McCoy teaches that typically, when operating in emulation mode, the native system is incapable of performing functions other than those of the terminal which is being emulated (column 1, lines 32-39) and his method provides a way of switching between running native and non-native functions. Further, both Duvall and McCoy are directed to the emulation of a non-native program module in a native environment and translating the non-native data into native data.

Duvall discloses virtual machine (VM) technology, which the Applicant discusses in its Background section on p. 7 and 8 of the Application. Duvall discloses a new addressing scheme for VMs to use to read/write from/to a "file" (rather than memory). Duvall does not disclose "limiting available memory to a range that is addressable." Rather, it discloses a re-definition and re-arrangement of the meaning of the bits in the existing and unmodified addressable range.

To make it clear that the "limiting" has the effect of reducing the range of available memory that a non-native program module may address, Applicant amends herein this claim in this manner:

limiting available memory to a range that is addressable by the non-native program module, that range of addressable memory being less than the available memory

Applicant submits that this amendment does not narrow the claim. Rather is clarifies its original meaning.

Moreover, the Office has not identified where Duvall discloses "non-native" program modules. Indeed, since Duvall discloses a VM model, then all program modules operating under a particular VM are presumptively native to that VM. If not, then an emulator would be necessary, but Duvall does not disclose an emulator.

While McCoy does disclose a nominal "emulator," it is not an emulation related to program modules being considered native or non-native. Rather, McCoy discloses a terminal emulation—that is, emulation of the operation of a "dumb" terminal connected to a host computer (e.g., mainframe computer).

The Office indicates the McCoy discloses an initiation of a program module based upon a determination of whether a program is native or non-native. It points to col. 5, lines 40-48, Fig. 3a, element 36a, which is reproduced here:

Keystrokes on the keyboard/display 35a are examined by the keystroke interpretation portion 36a of the emulation program to determine whether a PC or a host function is required. Program block 36a is responsive to the selected mode. In the PC mode, the keystrokes are handled by block 37a as normal keyboard commands or data. In the emulation mode, the keystrokes representing the host keys are passed to the host processor via the host emulator 31a.

1 However, this particular cited portion (and McCoy as a whole) are focused
2 on determining from whence input (e.g., keystrokes) is received and processing
3 them accordingly. The first sentence of the passage above says, "Keystrokes...are
4 examined...to determine whether a PC or a host function is required." Applicant
5 respectfully submits that this is not equivalent to "determining whether an
6 initiating program module is a native or non-native."

7 Indeed, Applicant submits that all of McCoy's program modules (include
8 the McCoy's terminal emulation program itself) are presumptively native. If they
9 were non-native, then they would not function on the PC absent an operating-
10 system based emulation program. However, McCoy does not disclose a such an
11 emulation program.

12 For the reasons given above, Applicant submits the combination of Duvall
13 and McCoy fail to disclose all of the elements and features of this claim.
14 Accordingly, Applicant asks the Office to withdraw its rejection of this claim.
15

16 No Motivation to Combine References

17 Furthermore, Applicant asserts that there is no motivation to combine the
18 teachings of Duvall and the teachings of McCoy.

19 As discussed above, Duvall describes an addressing scheme for accessing
20 files in a VM environment. However, McCoy describes "dumb" terminal
21 emulation on a PC.

22 Applicant submits that there is no suggestion, teaching, or reason given by
23 one reference that would motivate one of ordinary skill in the art at the time of the
24 invention (hereinafter, "OOSA") to combine it with the teachings of the other
25

1 reference. More importantly, Applicant submits that the Office has not provided
 2 any objective evidence showing why OOSA would be motivated to combine the
 3 teachings of the two references.

4 **Duvall** says nothing that would motivate OOSA to look towards **Chipman**
 5 and combine their teachings. Likewise, **McCoy** says nothing that would motivate
 6 OOSA to look towards **Duvall** and combine their teachings.

7 Accordingly, Applicant submits that OOSA would not be motivated to
 8 combine the VM file-access I/O addressing scheme of **Duvall** with the “dumb”
 9 terminal emulation of **McCoy**. Accordingly, Applicant asks the Office to
 10 withdraw its rejection of this claim.

11
 12 Claims 30-33

13 These claims ultimately depend upon independent claim 29. As discussed
 14 above, claim 29 is allowable.

15 In addition to its own merits, each of these dependent claims is allowable
 16 for the same reasons that its base claim is allowable. Applicant submits that the
 17 Office withdraw the rejection of each of these dependent claims because its base
 18 claim is allowable.
 19
 20
 21
 22
 23
 24
 25

421 West Riverside, Suite 500
 Spokane, WA 99201
 P: 509.324.9256
 F: 509.323.8979
 www.lee&hayes.com

lee & hayes

Dependent Claims

In addition to its own merits, each dependent claim is allowable for the same reasons that its base claim is allowable. Applicant submits that the Office withdraw the rejection of each dependent claim where its base claim is allowable.

Conclusion

All pending claims are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that prevent issuance of this application, the Office is urged to contact the undersigned attorney before issuing a subsequent Action.

Dated: 4-29-05By: 

Respectfully Submitted,

Kasey C. Christie
Reg. No. 40589
(509) 324-9256 x232
kasey@leehayes.com
www.leehayes.com

421 West Riverside, Suite 500
Spokane, WA 99201
P: 509.324-9256
F: 509.323-8979
www.leehayes.com
lee & hayes

Serial No.: 09/847,535
Atty Docket No.: MS1-665us
RESPONSE TO OFFICE ACTION DATED 11/30/2004

31

0408051508 G:\MS1-0865us\MS1-865us m01.doc

atty: Kasey C. Christie